

IMPLEMENTACIÓ DATAWAREHOUSE, DATAMART AMB JETTY, POSTGRESQL I WORKBECNH

Fernando Padilla Sánchez Oriol Riu Gispert
BASE DE DADES AVANÇADES Universitat de Barcelona

INDEX

- 1. Objectius**
- 2. Implementació**
- 3. Arxius modificats**
- 4. Import, Export database**
- 5. Problemes**

OBJECTIUS

L'objectiu d'aquesta segona pràctica és la implementació de un datawarehouse similar al cadastre que podem trobar a qualsevol ciutat. Per explotar les dades hem definit dos models d'estrella, un per a immobles i un altre per al sòl.

Les eines emprades per a realitzar el treball són:

- Jetty, un java-servlet de funcionament similar al vist en TomCat.
- Workbench, software per a definir cubs que representen una taula de fets, amb dimensions i mètriques.
- PostgreSQL (concretament utilitzem pgAdmin III per a muntar la base de dades i afegir dades).

IMPLEMENTACIÓ

PART CODI

Donem per suposat que les taules SQL s'han fet. Disponibles al zip entregat

La pràctica inclou diferents arxius que han de ser modificats per tal d'obtenir una representació segons les dades que hem introduït.

A la carpeta jetty/webapps.demo/mondrian/web-inf/queries modifiquem els .jsp que necessitem per a incloure les nostres consultes.

Per al model estrella del immoble ha modificat i afegit el següents arxius:

1. Data.jsp
2. Persones_fisiques.jsp
3. Persones_juridiques.jsp
4. Proves.jsp
5. Sol.jsp
6. Inmobles.jsp

En el cas del model estrella tenim els següents arxius:

1. Data_sol.jsp
2. Persones_fisiques_sol.jsp
3. Persones_juridiques_sol.jsp
4. Proves_juridiques_sol.jsp
5. Us_sol.jsp

Exemple de consulta

select

{[Measures].[persones Juridiques]} on columns,

{[Sol].[All Sol]} on rows

from [Inmoble]

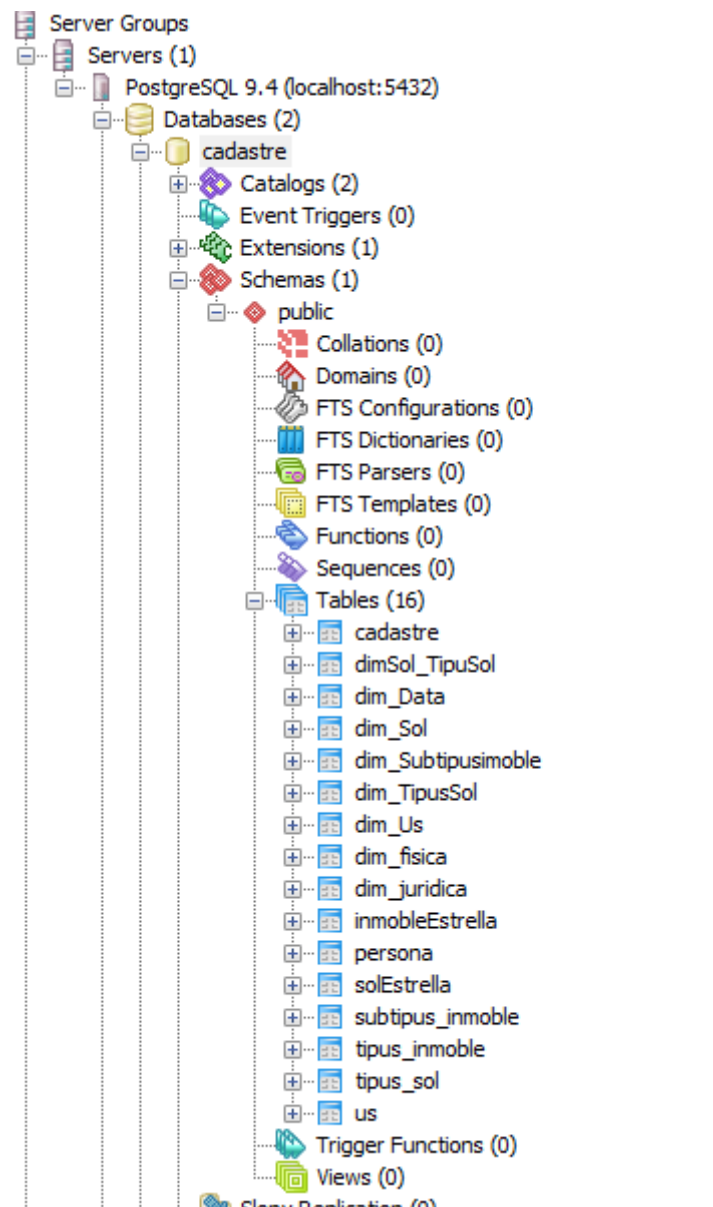
Configuració jsp

```
<jp:mondrianQuery id="query01" jdbcDriver="org.postgresql.Driver"  
jdbcUrl="jdbc:postgresql://localhost:5432/cadaestre?us
```

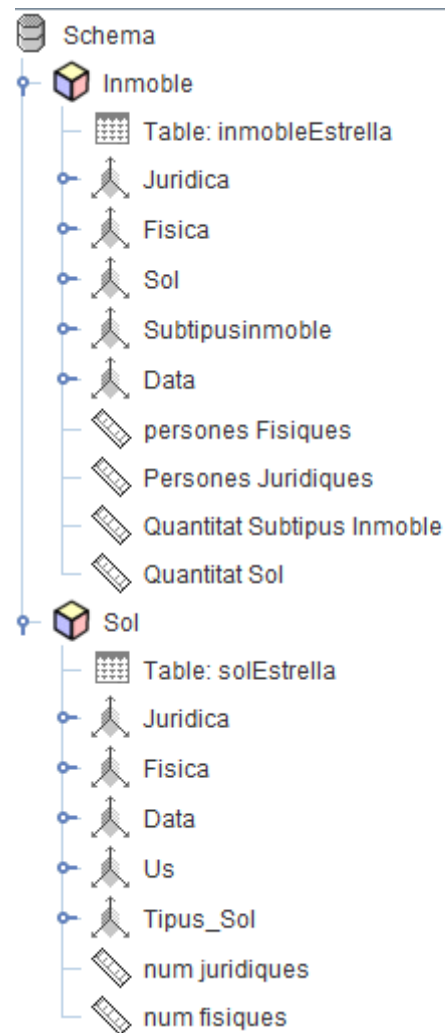
Indiquem el id de la query, el driver que utilitzarem, en aquest cas el corresponent al PostgreSQL, la direcció jdbc per a connectar workbench amb postgresql i el hostname amb el port on es troba la base de dades.

PART INTERFÍCIE I DADES

PostgreSQL ens permet muntar una base de dades amb dades dummy, les taules de fets, i les taules corresponents a cada dimensió que hem definit.



WORKBENCH



Podem observar que existeixen dos cubs, un per immoble i un altre per sòl, amb les dimensions respectives i les mètriques.

L'esquema dels nostres cubs es troba dins de la carpeta
jetty/webapss.demo/mondrian/web-inf/quèries/inmoble.xml

IMPORT, EXPORT DATABASE

LINUX

Importar

```
sudo -U usuari psql bbdd < arxiu.sql
```

Exportar

```
pg_dump bbdd -U usuari > arxiu.sql
```

WINDOWS

Importar

```
psql.exe -U usuari -d database -f arxiu.sql
```

Exportar

```
Pg_dump -U usuari bbdd > arxiu.sql
```

PROBLEMES

De problemes ens hem trobat diversos i de diferents origen, ja sigui per entorns de treball diferents, Windows y Linux, com incompatibilitats de software fins a errors de semàntica entre PostgreSQL i Workbench.

1. En Windows l'exportació de la base de dades no ha donat problemes en cap moment. No obstant, a Linux, quan el tamany de la BBDD ha arribat a un punt de considerar-se gran, pensem que Linux o triga moltíssim en generar l'arxiu SQL o simplement hi ha algun problema que nosaltres no podem arreglar. La solució ha sigut la més simple, copiar per altres vies la BBDD.
2. Quan muntant la base de dades a postgres, un dels errors més comuns ha sigut la no existència de PK en algunes dimensions. Això provocava que no poguéssim implementar correctament la BBDD.
3. Constantment jetty ens retorna errors de columnes no existents en la taula de fets de sòl, encara que sabem amb certesa que les dades són existents. La taula de fets de immoble no llença cap error en les mateixes consultes.
4. Quan observem les dades després de que jetty hi ha hagi mostrat els resultats, en certs casos obtenim dades de menys o dades que no són representatives del que hi ha a la BBDD.
5. En el cas de All Data, per mostrar per any, mes i dia, ens dona un error a la consulta select. Indica que no hi ha fills presents.
6. Alhora de configurar els arxius .jsp, jetty ens retorna errors a les consultes MDX, del tipus; no s'ha tancat la línia amb { } o Sol no existeix al cub.
7. Donada la complexitat de la implementació dels cubs, el resultat final no és un cadastre complet. Exemple final a continuació:

1. Tipologia dels locals cadastrals. Any 2002

1.1. Tipologia per districtes

Districte	TOTAL LOCALS			
	CADASTRALS	Nombre de locals habitatge	Nombre de locals aparcaments	Altres locals
BARCELONA	1.275.953	784.576	269.686	221.691
1. Ciutat Vella	80.856	58.078	4.755	18.023
2. Eixample	243.664	140.246	54.492	48.926
3. Sants-Montjuïc	139.812	84.148	28.349	27.315
4. Les Corts	79.637	39.622	27.487	12.528
5. Sarrià-St Gervasi	146.679	73.812	45.628	27.239
6. Gràcia	103.608	67.475	19.477	16.656
7. Horta-Guinardó	124.243	83.309	22.698	18.236
8. Nou Barris	104.058	75.202	15.068	13.788
9. Sant Andreu	99.637	63.450	20.092	16.095
10. Sant Martí	153.759	99.234	31.640	22.885

Una de les nostres consultes que serveix com a exemple.

Us tipus sol



	Tipus_Sol		
Us	-All Tipus Sol	+un sol amb problemes	+un sol en construcció
-All Us	3	1	2
-us comercial	2		2
1	2		2
+us privat	1	1	

CONCLUSIONS

Hem trobat que la practica pot ser molt interessant, tot i que amb una manca de temps important per poder finalitzar-la correctament, ja que molts dies del principi de la pràctica només s'han aprofitat per dissenyar els cubs i no s'ha pogut avançar. Realment amb 5 dies passar d'un model relacional a un datawarehouse de forma correcta i eficient sent el primer cop que es treballa amb aquests nivells de bases de dades es molt poc temps.

Tot i això al final creiem que hem aconseguit una pràctica bona.